Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	329953	(table or tuple or array) with (partition$2 or subtable or portion or part)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:08
L2	1718	(organiz$3 with (row or record$3)) and 1	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:09
L3	46	2 and (hash with (merg$3 or sort$3))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:11
L4	31	3 and @ad<"20011207"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:38
L5	4	4 and ((partition$2 or subtable or portion or part) with (merg$3 or sort$3))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:16
L6	30	4 and (hash with table)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:25
L7	2	4 and (hash with file)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:17
L8	16	4 and (track$3 with (row or record$3))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:26

L9	2	4 and (track$3 with hash)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:34
L10	1	4 and (707/200).ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:34
L11	0	4 and (707/1).ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:34
L12	0	4 and (707/7).ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:34
L13	23	4 and (locat$3 with (row or record$3))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:36
L14	0	12 and (file with partition)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:36
L15	0	12 and ((merg$3 or sort$3) with partition)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/08 15:38

Web   Images   Groups   News   Froogle   Local^{New!}   **more »**

Google |partition merge hash table reorganizing rows | Search | Advanced Search
Preferences
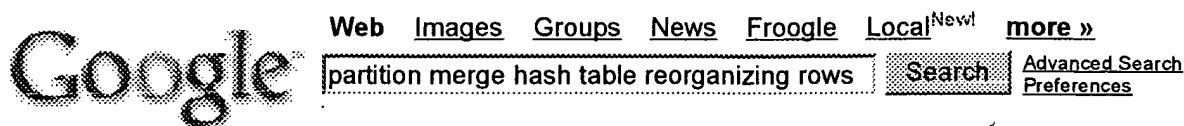
Web   Results **1 - 10** of about **592** for <u>partition</u> <u>merge</u> <u>hash</u> <u>table</u> <u>reorganizing rows</u>. (**0.16** seconds)

[PDF] <u>Oracle8 i Enterprise Edition Partitioning Option</u>
File Format: PDF/Adobe Acrobat - <u>View as HTML</u>
Equality predicates on **hash** partitioned. of **partition** elimination. equi-partitioned
... per-**table** basis. •. A new **partition merge** operation now. complements ...
www.oracle.com/technology/ products/oracle8i/pdf/ent_part_fo.pdf - <u>Similar pages</u>

[PDF] <u>Oracle Partitioning Option</u>
File Format: PDF/Adobe Acrobat - <u>View as HTML</u>
rows to particular **partitions**. The range partitioning method is particularly ...
Hash partitioning can be used to spread data. evenly among **partitions** ...
www.oracle.com/technology/ products/oracle8i/pdf/part_ds.pdf - <u>Similar pages</u>

<u>Oracle9i Administration and Management - training</u>
... **rows**, imbalanced freelists • **Reorganizing Tables** using Export and Import ...
Merge join • **Hash** Join • Nested Loop join • Advanced SQL operators ...
www.dba-oracle.com/cou_intro-oracle.htm - 67k - <u>Cached</u> - <u>Similar pages</u>

<u>%Title% (Archived Microsoft SQL Server 7.0 Technical Articles)</u>
... SQL Server 7.0 adds support for **"merge"** and **"hash"** join techniques. ...
Some **tables** are so large that it may make sense to **"partition"** them into smaller ...
msdn.microsoft.com/archive/en-us/ dnarsql7/html/deploybus_devoverview.asp?FRAME=true - 102k -
<u>Cached</u> - <u>Similar pages</u>

[PDF] <u>The DBA Corner: Partitioning Demystified</u>
File Format: PDF/Adobe Acrobat
join the other **tables** to these **partitions** in parallel and then **merge** the ...
Instead of **reorganizing** the whole **table**, a **partition** could be reorganized. ...
www.nyoug.org/200212nanda.pdf - <u>Similar pages</u>

<u>Dwaine Snow's Thoughts on DB2: October 2005</u>
For each **row** in the probe **table** the same **hashing** algorithm is applied to the join
... If the **partition** that matches the probe **table row** is in memory the ...
dsnowondb2.blogspot.com/ 2005_10_01_dsnowondb2_archive.html - 65k - <u>Cached</u> - <u>Similar pages</u>

<u>HP 3000 Manuals</u>
When a **HASH table** is defined, the allocation and formatting of the **HASH** pages
... GENERATE **PARTITION** - generates ALLBASE/SQL CREATE **PARTITION** commands to ...
docs.hp.com/cgi-bin/doc3k/B3021690189.13992/78 - 36k - <u>Cached</u> - <u>Similar pages</u>

[PDF] <u>Efficient Bulk Deletes in Relational Databases</u>
File Format: PDF/Adobe Acrobat - <u>View as HTML</u>
hybrid **hash** join which would **partition** the **table**/index is not. viable. ...
the bulk deletion for **table** R. **Reorganizing table** R, however, ...
www.dbis.ethz.ch/research/publications/39.pdf - <u>Similar pages</u>

[PDF] <u>Parallel Database Systems: The Future of High Performance Database ...</u>
File Format: PDF/Adobe Acrobat - <u>View as HTML</u>
is compared against the main-memory **hash table** for the A **partition**. If there is
a match, ... sort-**merge** join, and **hash** join algorithms are provided. ...

www.cs.wisc.edu/~dewitt/includes/paralleldb/cacm.pdf - Similar pages

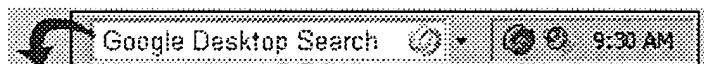[PDF] A Dynamic Perfect **Hash** Function Defined by an Extended **Hash** ...
File Format: PDF/Adobe Acrobat - View as HTML
table to. define or modify the **hash** functions. The auxiliary. **table** is referred
... **table**. An entry. in **row** h and column k is a singleton. if there ...
www.vldb.org/conf/1984/P245.PDF - Similar pages

Goooooooooogle ▶
Result Page: **1** 2 3 4 5 6 7 8 9 10   **Next**

Google Desktop Search ◎ ▾ | 🌐 ◎ 9:30 AM
Free! Instantly find your email, files, media and web history. Download now.

partition merge hash table reorganiz   Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - About Google

IEEE Xplore
RELEASE 2.1

Welcome United States Patent and Trademark Office

Search Results   BROWSE   SEARCH   IEEE XPLORE GUIDE

Results for **"(partition merge hash table file row database index<in>metadata)"**   ☑e-mail
Your search matched **0** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» Search Options

View Session History

New Search

Modify Search

(partition merge hash table file row database index<in>metadata)   [»]

☐ Check to search only within this results set

Display Format: ⦿ Citation   ◯ Citation & Abstract

» Key

|	
IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistan
search.

Help   Contact Us   Privacy & :

Indexed by
Inspec

IEEE Xplore® RELEASE 2.1

BROWSE   SEARCH   IEEE XPLORE GUIDE

Search Results

Results for **"(partition merge hash table<in>metadata)"**
Your search matched **0** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

☑ e-mail

» Search Options

View Session History

New Search

Modify Search

(partition merge hash table<in>metadata) [»]

☐ Check to search only within this results set

» Key

Display Format: ⦿ Citation ○ Citation & Abstract

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistan
search.

P☉RTAL

USPTO
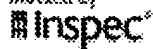
Search: ◉ The ACM Digital Library   ○ The Guide

partition merge hash table file row reorganizing

THE ACM DIGITAL LIBRARY

☞ Feedback Report a problem Satisfaction survey

Terms used **partition** **merge** **hash** **table** **file** **row** **reorganizing**

Found **21,076** of **166,357**

| Sort results by | relevance ▾ |
| Display results | expanded form ▾ |

◆ Save results to a Binder

? Search Tips

☐ Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale ☐ ▭ ▬ ▬ ▮

1 **Query evaluation techniques for large databases**

Goetz Graefe

June 1993 **ACM Computing Surveys (CSUR)**, Volume 25 Issue 2

Publisher: ACM Press

Full text available: 📄 pdf(9.37 MB)   Additional Information: full citation, abstract, references, citings, index terms, review

Database management systems will continue to manage large data volumes. Thus, efficient algorithms for accessing and manipulating large sets and sequences will be required to provide acceptable performance. The advent of object-oriented and extensible database systems will not solve this problem. On the contrary, modern data models exacerbate the problem: In order to manipulate large sets of complex objects as efficiently as today's database systems manipulate simple records, query-processi ...

Keywords: complex query evaluation plans, dynamic query evaluation plans, extensible database systems, iterators, object-oriented database systems, operator model of parallelization, parallel algorithms, relational database systems, set-matching algorithms, sort-hash duality

2 **Multidimensional access methods**

Volker Gaede, Oliver Günther

June 1998 **ACM Computing Surveys (CSUR)**, Volume 30 Issue 2

Publisher: ACM Press

Full text available: 📄 pdf(1.05 MB)   Additional Information: full citation, abstract, references, citings, index terms

Search operations in databases require special support at the physical level. This is true for conventional databases as well as spatial databases, where typical search operations include the point query (find all objects that contain a given search point) and the region query (find all objects that overlap a given search region). More than ten years of spatial database research have resulted in a great variety of multidimensional access methods to support ...

Keywords: data structures, multidimensional access methods

3

Comparison of access methods for time-evolving data

Betty Salzberg, Vassilis J. Tsotras
June 1999 **ACM Computing Surveys (CSUR)**, Volume 31 Issue 2
Publisher: ACM Press

Full text available: pdf(529.53 KB)   Additional Information: full citation, abstract, references, citings, index terms

This paper compares different indexing techniques proposed for supporting efficient access to temporal data. The comparison is based on a collection of important performance criteria, including the space consumed, update processing, and query time for representative queries. The comparison is based on worst-case analysis, hence no assumptions on data distribution or query frequencies are made. When a number of methods have the same asymptotic worst-case behavior, features in the methods tha ...

Keywords: I/O performance, access methods, structures, temporal databases

4 Access methods for text

Chris Faloutsos
March 1985 **ACM Computing Surveys (CSUR)**, Volume 17 Issue 1
Publisher: ACM Press

Full text available: pdf(2.59 MB)   Additional Information: full citation, abstract, references, citings, index terms, review

This paper compares text retrieval methods intended for office systems. The operational requirements of the office environment are discussed, and retrieval methods from database systems and from information retrieval systems are examined. We classify these methods and examine the most interesting representatives of each class. Attempts to speed up retrieval with special purpose hardware are also presented, and issues such as approximate string matching and compression are discussed. A quali ...

5 Partially preemptible hash joins

Hwee Hwa Pang, Michael J. Carey, Miron Livny
June 1993 **ACM SIGMOD Record , Proceedings of the 1993 ACM SIGMOD international conference on Management of data SIGMOD '93**, Volume 22 Issue 2
Publisher: ACM Press

Full text available: pdf(1.42 MB)   Additional Information: full citation, abstract, references, citings, index terms

With the advent of real-time and goal-oriented database systems, priority scheduling is likely to be an important feature in future database management systems. A consequence of priority scheduling is that a transaction may lose its buffers to higher-priority transactions, and may be given additional memory when transactions leave the system. Due to their heavy reliance on main memory, hash joins are especially vulnerable to fluctuations in memory availability. Previous studies have propose ...

6 Associative searching in multiple storage units

C. T. Wu, Walter A. Burkhard
March 1987 **ACM Transactions on Database Systems (TODS)**, Volume 12 Issue 1
Publisher: ACM Press

Full text available: pdf(1.83 MB)   Additional Information: full citation, abstract, references, citings, index terms, review

A file maintenance model, called the multiple random access storage units model, is introduced. Storage units can be accessed simultaneously, and the parallel processing of an associative query is achieved by distributing data evenly among the storage units. Maximum parallelism is obtained when data satisfying an associative query are evenly distributed for every possible query. An allocation scheme called M-cycle allocation is proposed to maintain large files of data on mu ...

7 Object-based and image-based object representations

Hanan Samet

June 2004 **ACM Computing Surveys (CSUR)**, Volume 36 Issue 2

Publisher: ACM Press

Full text available: pdf(1.05 MB)   Additional Information: full citation, abstract, references, index terms

An overview is presented of object-based and image-based representations of objects by their interiors. The representations are distinguished by the manner in which they can be used to answer two fundamental queries in database applications: (1) Feature query: given an object, determine its constituent cells (i.e., their locations in space). (2) Location query: given a cell (i.e., a location in space), determine the identity of the object (or objects) of which it is a member as well as the re ...

Keywords: Access methods, R-trees, feature query, geographic information systems (GIS), image space, location query, object space, octrees, pyramids, quadtrees, space-filling curves, spatial databases

8 Concepts and capabilities of a database computer\

Jayanta Banerjee, David K. Hsiao, Richard I. Baum

December 1978 **ACM Transactions on Database Systems (TODS)**, Volume 3 Issue 4

Publisher: ACM Press

Full text available: pdf(2.79 MB)   Additional Information: full citation, abstract, references, citings, index terms

The concepts and capabilities of a database computer (DBC) are given in this paper. The proposed design overcomes many of the traditional problems of database system software and is one of the first to describe a complete data-secure computer capable of handling large databases. This paper begins by characterizing the major problems facing today's database system designers. These problems are intrinsically related to the nature of conventional hardware and can only be solved by i ...

Keywords: clustering, content-addressable memory, database computers, keywords, mass memory, performance, security, structure memory

9 Conference abstracts

January 1977 **Proceedings of the 5th annual ACM computer science conference**

Publisher: ACM Press

Full text available: pdf(3.14 MB)   Additional Information: full citation, abstract, index terms

One problem in computer program testing arises when errors are found and corrected after a portion of the tests have run properly. How can it be shown that a fix to one area of the code does not adversely affect the execution of another area? What is needed is a quantitative method for assuring that new program modifications do not introduce new errors into the code. This model considers the retest philosophy that every program instruction that could possibly be reached and tested from the ...

10 The state of the art in distributed query processing

Donald Kossmann

December 2000 **ACM Computing Surveys (CSUR)**, Volume 32 Issue 4

Publisher: ACM Press

Full text available: pdf(455.39 KB)   Additional Information: full citation, abstract, references, citings, index terms

Distributed data processing is becoming a reality. Businesses want to do it for many reasons, and they often must do it in order to stay competitive. While much of the infrastructure for distributed data processing is already there (e.g., modern network technology), a number of issues make distributed data processing still a complex undertaking: (1) distributed systems can become very large, involving thousands of heterogeneous sites including PCs and mainframe server machines; (2) the stat ...

Keywords: caching, client-server databases, database application systems, dissemination-based information systems, economic models for query processing, middleware, multitier architectures, query execution, query optimization, replication, wrappers

11 An incremental access method for ViewCache: concept, algorithms, and cost analysis
Nicholas Roussopoulos
September 1991 **ACM Transactions on Database Systems (TODS)**, Volume 16 Issue 3
Publisher: ACM Press

Full text available: pdf(1.71 MB)   Additional Information: full citation, abstract, references, citings, index terms, review

A ViewCache is a stored collection of pointers pointing to records of underlying relations needed to materialize a view. This paper presents an Incremental Access Method (IAM) that amortizes the maintenance cost of ViewCaches over a long time period or indefinitely. Amortization is based on deferred and other update propagation strategies. A deferred update strategy allows a ViewCache to remain outdated until a query needs to selectively or ...

Keywords: terms

12 Industrial sessions: big data: Automating physical database design in a parallel database
Jun Rao, Chun Zhang, Nimrod Megiddo, Guy Lohman
June 2002 **Proceedings of the 2002 ACM SIGMOD international conference on Management of data**
Publisher: ACM Press

Full text available: pdf(1.38 MB)   Additional Information: full citation, abstract, references, citings, index terms

Physical database design is important for query performance in a shared-nothing parallel database system, in which data is horizontally partitioned among multiple independent nodes. We seek to automate the process of data partitioning. Given a workload of SQL statements, we seek to determine automatically how to partition the base data across multiple nodes to achieve overall optimal (or close to optimal) performance for that workload. Previous attempts use heuristic rules to make those decision ...

13 Scalable feature selection, classification and signature generation for organizing large text databases into hierarchical topic taxonomies
Soumen Chakrabarti, Byron Dom, Rakesh Agrawal, Prabhakar Raghavan
August 1998 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 7 Issue 3
Publisher: Springer-Verlag New York, Inc.

Full text available: pdf(281.37 KB)   Additional Information: full citation, abstract, citings, index terms

We explore how to organize large text databases hierarchically by topic to aid better searching, browsing and filtering. Many corpora, such as internet directories, digital

libraries, and patent databases are manually organized into topic hierarchies, also called *taxonomies*. Similar to indices for relational data, taxonomies make search and access more efficient. However, the exponential growth in the volume of on-line textual information makes it nearly impossible to maintain such taxono ...

14 Compiler-directed run-time monitoring of program data access

Chen Ding, Yutao Zhong

June 2002 **ACM SIGPLAN Notices , Proceedings of the 2002 workshop on Memory system performance MSP '02**, Volume 38 Issue 2 supplement

Publisher: ACM Press

Full text available: pdf(1.40 MB)   Additional Information: full citation, abstract, references, citings

Accurate run-time analysis has been expensive for complex programs, in part because most methods perform on all a data. Some applications require only partial reorganization. An example of this is off-loading infrequently used data from a mobile device. Complete monitoring is not necessary because not all accesses can reach the displaced data. To support partial monitoring, this paper presents a framework that includes a source-to-source C compiler and a run-time monitor. The compiler inserts ru ...

15 Technical reports

SIGACT News Staff

January 1980 **ACM SIGACT News**, Volume 12 Issue 1

Publisher: ACM Press

Full text available: pdf(5.28 MB)   Additional Information: full citation

16 External memory algorithms and data structures: dealing with **massive data**

Jeffrey Scott Vitter

June 2001 **ACM Computing Surveys (CSUR)**, Volume 33 Issue 2

Publisher: ACM Press

Full text available: pdf(828.46 KB)   Additional Information: full citation, abstract, references, citings, index terms

Data sets in large applications are often too massive to fit completely inside the computers internal memory. The resulting input/output communication (or I/O) between fast internal memory and slower external memory (such as disks) can be a major performance bottleneck. In this article we survey the state of the art in the design and analysis of external memory (or EM) algorithms and data structures, where the goal is to exploit locality in order to reduce the I/O costs. We consider a varie ...

Keywords: B-tree, I/O, batched, block, disk, dynamic, extendible hashing, external memory, hierarchical memory, multidimensional access methods, multilevel memory, online, out-of-core, secondary storage, sorting

17 A database approach for managing VLSI design data

Randy H. Katz

January 1982 **Proceedings of the 19th conference on Design automation**

Publisher: IEEE Press

Full text available: pdf(830.15 KB)   Additional Information: full citation, abstract, references, citings, index terms

We describe an approach to managing information about VLSI designs, founded upon database system methods. A database component provides a low-level flat-file interface to stored data. Built on top is a design data management system, supporting the

hierarchical construction of a design from primitive cells, and organizing data about alternative design representations and versions. Programs to provide a tailored interface to design data are also provided. The system simplifies the rapid const ...

18 Theoretical aspects: Distribution-dependent hashing functions and their characteristics
R. F. Deutscher, P. G. Sorenson, J. P. Tremblay
May 1975 **Proceedings of the 1975 ACM SIGMOD international conference on Management of data**
Publisher: ACM Press
Full text available: pdf(1.46 MB)   Additional Information: full citation, abstract, references, citings

In this paper procedures are studied for storing, accessing, updating, and reorganizing data in large files whose organization is direct, an organization used when a fast response time is required. "Distribution-dependent" hashing functions and the division method are compared as methods of indirect addressing. "Distribution-dependent" hashing functions are characterized. These hashing functions generate addresses from a set of keys by using knowledge of the distribution of that key set within th ...

Keywords: data base design, data base reorganization, hashing functions

19 Data structures for efficient broker implementation
Anthony Tomasic, Luis Gravano, Calvin Lue, Peter Schwarz, Laura Haas
July 1997 **ACM Transactions on Information Systems (TOIS)**, Volume 15 Issue 3
Publisher: ACM Press
Full text available: pdf(316.45 KB)   Additional Information: full citation, abstract, references, citings, index terms, review

With the profusion of text databases on the Internet, it is becoming increasingly hard to find the most useful databases for a given query. To attack this problem, several existing and proposed systems employ brokers to direct user queries, using a local database of summary information about the available databases. This summary information must effectively distinguish relevant databases and must be compact while allowing efficient access. We offer evidence that one broker, GlOSS
Keywords: GlOSS, broker architecture, broker performance, distributed information, grid files, partitioned hashing

20 The Quadtree and Related Hierarchical Data Structures
Hanan Samet
June 1984 **ACM Computing Surveys (CSUR)**, Volume 16 Issue 2
Publisher: ACM Press
Full text available: pdf(4.87 MB)   Additional Information: full citation, references, citings, index terms